

**In the Claims:**

Please amend claims 1, 6, 7, 13, 14, 17, 18, 19, 21, 26, 27, 32, 33, and 37-42, and please cancel claims 3-5, 16, 23-25, and 34-36, as indicated below:

1. (Currently amended) A system, comprising:

a processor; and

a memory comprising program instructions, wherein the program instructions are executable by the processor to implement:

file system software configured to assign and migrate data in a multi-class file system comprising a hierarchy of storage classes, wherein the migrated data remains online within the multi-class file system, wherein one or more of the storage classes store data that is not modifiable by applications while the data is on the one or more storage classes, and wherein one or more others of the storage classes store data that is modifiable by applications while the data is on the one or more storage classes; and

an application configured to perform an operation on data stored in the multi-class file system, wherein the operation that requires stable data, wherein, to perform the operation ~~on the one or more storage classes that store data that is not modifiable by applications while the data is on those storage classes,~~ the application is configured to:

perform the operation on at least one of the one or more storage classes that store data that is not modifiable without using a split mirror of the at least one of the one or more storage

classes; and

perform the operation on at least one of the one or more other storage classes that store data that is modifiable using a split mirror of the at least one of the one or more other storage classes.

2. (Original) The system as recited in claim 1, wherein the operation is a backup of the storage classes.

3. (Canceled)

4. (Canceled)

5. (Canceled)

6. (Currently amended) The system as recited in claim 1, wherein the one or more storage classes that store data that is not modifiable by applications while the data is on those storage classes are configured to be write locked, and wherein, to perform the operation on at least one of the one or more storage classes that store data that is not modifiable without using a split mirror of the at least one of the one or more storage classes, the application is further configured to, for each write-locked storage class on which the operation is performed:

examine a write lock of ~~each~~ the write-locked storage class to determine if the write-locked storage class could have been written to during the operation on that storage class; and

if said examine a write lock of the write-locked storage class determines the storage class has could have been written to during the operation on ~~the~~ that storage class, retry the operation for the write-locked storage class.

7. (Currently amended) The system as recited in claim 1, wherein the one or more storage classes that store data that is not modifiable by applications while the data is on those storage classes are configured to be write locked, and wherein the application is further configured to block the file system software from enabling ~~the~~ a write-locked storage class for writing for the duration of the operation.

8. (Original) The system as recited in claim 1, wherein the data includes files or portions of files.

9. (Original) The system as recited in claim 1, wherein the data comprises one or more of application data and file system metadata.

10. (Original) The system as recited in claim 1, wherein the storage classes are ordered in the hierarchy according to one or more characteristics from a highest storage class to a lowest storage class.

11. (Original) The system as recited in claim 10, wherein the one or more characteristics include one or more of performance and cost.

12. (Original) The system as recited in claim 1, wherein the storage classes are ordered in the hierarchy of storage classes according to performance characteristics from a highest storage class comprising one or more high-performance storage devices to a lowest storage class comprising one or more low-performance storage devices.

13. (Currently amended) A system, comprising:

a processor; and

a memory comprising program instructions, wherein the program instructions are executable by the processor to implement:

file system software configured to assign and migrate data in a multi-class file system comprising a hierarchy of storage classes, wherein migrated data remains online within the multi-class file system; and

an application configured to perform an operation on the storage classes that requires stable data, wherein, to perform the operation, the application is configured to:

perform the operation on at least one of the storage classes without using a split mirror; and

perform the operation on at least one other of the storage classes using a split mirror.

14. (Currently amended) A system, comprising:

a plurality of storage devices;

a host system configured to couple to the plurality of storage devices via a network, wherein the host system comprises:

file system software configured to assign and migrate data in a multi-class file system comprising a hierarchy of storage classes, wherein migrated data remains online within the multi-class file system, wherein one or more of the storage classes store data that is not modifiable by applications while the data is on the one or more storage classes, and wherein one or more others of the storage classes store data that is modifiable by applications while the data is on the one or more storage classes; and

an application configured to perform an operation on data stored in the multi-class file system, wherein the operation that requires stable data, wherein, to perform the operation on the one or more storage classes that store data that is not modifiable by applications while the data is on those storage classes, the application is configured to:

perform the operation on at least one of the one or more storage classes that store data that is not modifiable without using a split mirror of the at least one of the one or more storage classes; and

perform the operation on at least one of the one or more other storage classes that store data that is modifiable using a split mirror of the at least one of the one or more other storage classes.

15. (Original) The system as recited in claim 14, wherein the operation is a backup of the storage classes.

16. (Canceled)

17. (Currently amended) The system as recited in claim 14, wherein the one or more storage classes that store data that is not modifiable by applications while the data is on those storage classes are configured to be write locked, and wherein, to perform the operation on at least one of the one or more storage classes that store data that is not modifiable without using a split mirror of the at least one of the one or more storage classes, the application is further configured to, for each write-locked storage class on which the operation is performed:

examine a write lock of ~~each~~ the write-locked storage class to determine if the write-locked storage class could have been written to during the operation on that storage class; and

if said examine a write lock of the write-locked storage class determines the storage class ~~has~~ could have been written to during the operation on ~~the~~ that storage class, retry the operation for the write-locked storage class.

18. (Currently amended) The system as recited in claim 14, wherein the one or more storage classes that store data that is not modifiable by applications while the data is on those storage classes are configured to be write locked, and wherein the application is further configured to block the file system software from enabling ~~the~~ a write-locked storage class for writing for the duration of the operation.

19. (Currently amended) A system, comprising:

software means for assigning and migrating data in a multi-class file system comprising a plurality of storage classes and for providing access to the data in the multi-class file system to one or more applications, wherein migrated data remains online within the multi-class file system, wherein one or more of the storage classes store data that is not modifiable by the applications while the data is on the one or more storage classes, and wherein one or more others of the storage classes store data that is modifiable by applications while the data is on the one or more storage classes; and

means for performing operations that require stable data on data stored in the multi-class file system, wherein said means for performing operations that require stable data comprise:

means for performing the operations on the one or more storage classes

~~that store data that is not modifiable by the applications while the data is on those storage classes;~~ without using a split mirror of the one or more storage classes; and

means for performing the operations on the one or more other storage classes that store data that is modifiable using a split mirror of the one or more other storage classes.

20. (Original) The system as recited in claim 19, further comprising means for blocking write access to the one or more storage classes that store data that is not modifiable for the duration of the operations.

21. (Currently amended) A method, comprising:

file system software assigning and migrating data in a multi-class file system comprising a hierarchy of storage classes, wherein migrated data remains online within the multi-class file system, wherein one or more of the storage classes store data that is not modifiable by applications while the data is on the one or more storage classes, and wherein one or more others of the storage classes store data that is modifiable by applications while the data is on the one or more storage classes; and

performing an operation on data stored in the multi-class file system, wherein the operation ~~that requires stable data,~~ wherein said performing the operation on the one or more storage classes that store data that is not modifiable by applications while the data is on those storage classes comprises;

performing the operation on at least one of the one or more storage classes that store data that is not modifiable without using a split mirror of the at least one of the one or more storage classes; and

performing the operation on at least one of the one or more other storage classes that store data that is modifiable using a split mirror of the at least one of the one or more other storage classes.

22. (Original) The method as recited in claim 21, wherein the operation is a backup of the storage classes.

23. (Canceled)

24. (Canceled)

25. (Canceled)

26. (Currently amended) The method as recited in claim 21, wherein the one or more storage classes that store data that is not modifiable by applications while the data is on those storage classes are write locked, and wherein said performing the operation on at least one of the one or more storage classes that store data that is not modifiable without using a split mirror of the at least one of the one or more storage classes comprises, for each write-locked storage class on which the operation is performed:

examining a write lock of ~~each~~ the write-locked storage class to determine if the write-locked storage class could have been written to during the operation on that storage class; and

if said examining a write lock of the write-locked storage class determines the storage class ~~has~~ could have been written to during the operation on the that storage class, retrying the operation for the write-locked storage class.

27. (Currently amended) The method as recited in claim 21, wherein the one or more storage classes that store data that is not modifiable by applications while the data is on those storage classes are configured to be write locked, and wherein the method



further comprising blocking the file system software from enabling the a write-locked storage class for writing for the duration of the operation.

28. (Original) The method as recited in claim 21, wherein the data includes files or portions of files.

29. (Original) The method as recited in claim 21, wherein the data comprises one or more of application data and file system metadata.

30. (Original) The method as recited in claim 21, wherein the storage classes are ordered in the hierarchy according to one or more characteristics from a highest storage class to a lowest storage class.

31. (Original) The method as recited in claim 21, wherein the storage classes are ordered in the hierarchy of storage classes according to performance characteristics from a highest storage class comprising one or more high-performance storage devices to a lowest storage class comprising one or more low-performance storage devices.

32. (Currently amended) A computer-accessible storage medium, comprising program instructions, wherein the program instructions are ~~configured~~ computer-executable to implement:

assigning and migrating data in a multi-class file system comprising a hierarchy of storage classes, wherein migrated data remains online within the multi-class file system, wherein one or more of the storage classes store data that is not modifiable by applications while the data is on the one or more storage classes, and wherein one or more others of the storage classes store data that is modifiable by applications while the data is on the one or more storage classes; and

performing an operation on data stored in the multi-class file system, wherein the

~~operation that requires stable data, wherein said performing the operation on the one or more storage classes that store data that is not modifiable by applications while the data is on those storage classes comprises;~~

~~performing the operation on at least one of the one or more storage classes that store data that is not modifiable without using a split mirror of the at least one of the one or more storage classes; and~~

~~performing the operation on at least one of the one or more other storage classes that store data that is modifiable using a split mirror of the at least one of the one or more other storage classes.~~

33. (Currently amended) The computer-accessible storage medium as recited in claim 32, wherein the operation is a backup of the storage classes.

34. (Canceled)

35. (Canceled)

36. (Canceled)

37. (Currently amended) The computer-accessible storage medium as recited in claim 32, wherein the one or more storage classes that store data that is not modifiable by applications while the data is on those storage classes are write locked, and wherein, in said performing the operation on at least one of the one or more storage classes that store data that is not modifiable without using a split mirror of the at least one of the one or more storage classes, the program instructions are further ~~configured~~ computer-executable to implement, for each write-locked storage class on which the operation is performed:

examining a write lock of ~~each~~ the write-locked storage class to determine if the

write-locked storage class could have been written to during the operation on that storage class; and

if said examining a write lock of the write-locked storage class determines the storage class ~~has~~ could have been written to during the operation on ~~the~~ that storage class, retrying the operation for the write-locked storage class.

38. (Currently amended) The computer-accessible storage medium as recited in claim 32, wherein the one or more storage classes that store data that is not modifiable by applications while the data is on those storage classes are configured to be write locked, and wherein the program instructions are further ~~configured~~ computer-executable to implement blocking file system software from enabling write access to ~~the~~ a write-locked storage class for the duration of the operation.

39. (Currently amended) The computer-accessible storage medium as recited in claim 32, wherein the data includes files or portions of files.

40. (Currently amended) The computer-accessible storage medium as recited in claim 32, wherein the data comprises one or more of application data and file system metadata.

41. (Currently amended) The computer-accessible storage medium as recited in claim 32, wherein the storage classes are ordered in the hierarchy according to one or more characteristics from a highest storage class to a lowest storage class.

42. (Currently amended) The computer-accessible storage medium as recited in claim 32, wherein the storage classes are ordered in the hierarchy of storage classes according to performance characteristics from a highest storage class comprising one or more high-performance storage devices to a lowest storage class comprising one or more low-performance storage devices.